International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 6 3867 - 3872

# Genetic Algorithm Approach for Implementation of Job Scheduling Problem

Sachin V. Solanki
Asstt. Professor, Department of Information Technology,
K.D.K. College of Engineering, Nagpur, India.

***Abstract-*** A job scheduling maps and schedules the virtual machine (VM) resources to physical machines (VM) for getting the finest mapping result to achieve the proper system load balance. Job scheduling system tries to find the best suitable schedule in a system for VMs and PMs, by considering various on time restrictions into concern. The ultimate goal of job scheduling is to schedule adaptable virtual machines to physical machines, getting a suitable order in order to enhance resource utility. This research paper proposes an approach in order to discuss a Job Scheduling problem to progress resource utility with the help of Genetic Algorithm (GA).

***Keywords** – Job Scheduling, Genetic Algorithm*

*****

## I. INTRODUCTION

The job scheduling problem has been discussed over the decade which is one of the toughest combinatorial optimization problems. This research paper proposes approach job Scheduling problem with the help of Genetic Algorithm (GA) and various results are also discussed on the basis of execution time and fitness values.

### *A. Job Scheduling Problem Overview*

The goal of this paper is to get an appropriate sequence by scheduling n VMs on m PMs. A job scheduling is a process that manages and maps the execution of jobs on the physical machines. It allocates appropriate jobs (in VMs) to PMs so the execution is often completed to satisfy objective functions imposed by users. Acceptable job scheduling will have important impact on the performance of the system. The common concern in scheduling jobs on distributed resources belongs to a category of issues called NP-hard issues. For this type of problems, it is tricky to get algorithms to create the optimal solution within polynomial time. Even if the task scheduling problem can be solved by using exhaustive search, the methods complexity for solving task scheduling is very high. To formulate the problem, consider *Jn* independent user jobs in VMs n={1,2,….N} on R*m* heterogeneous PMs m={1,2,….,M} with an objective of minimizing the completion time and utilizing the resources effectively. Any job *Jn* has to be processed in PM resource R*m*, until completion. The design and implementation of such a scheduling system is the matter of concern of this research paper.

## II. REVIEW OF LITERATURE

Job scheduling system selects the best appropriate jobs in a Cloud for Cloud computing users' requests, by considering various parameters constraints. Many research studies in Grid Computing can be applied directly in Clouding Computing environment. Research Papers [1-7] discussed a better outlook for the roles of job scheduling in a Grid computing situation. The presented topologies of job scheduling system in Cloud or Grid are divided into centralized and decentralized schedulers [1]. Zhiguang, S. and L. Chuang [2] focused a brief explanation of a modeling and performance assessment of hierarchical job scheduling, [3] discussed an iterative scheduling algorithms on the grids. Paper [4] introduced a new stochastic algorithm for QoS-constrained workflows job scheduling in a web service-oriented grid.

Several academic researchers started to work on the QoS of job scheduling system; that can be viewed in references [6-10] that set ahead the approach of QoS performance analysis for Cloud Computing services with dynamic scheduling system. Paper [11] discussed how the diversity of jobs characteristics such as unstructured/unorganized arrival of jobs and priorities, could lead to inefficient job allocation. Reference [15] presented a technique for job allocation for data processing services over the cloud considering amongst others the processing power, and memory requirements.

GA is able to narrow the search area around the required decision in a short time. However, because of stochastic characteristic of search strategy, completing the task can take considerable amount of time. Moreover, in scheduling tasks the initial information is represented as sets of discrete elements, which are connected with each other in non-trivial way. A number of studies have been devoted to methods of increasing of GA efficiency. Paper [20] represents combinations of GA and "traditional" search techniques. GAN Guo-ning, HUANG Ting-Iei, GAO Shuai [21] introduces an optimized method for task scheduling foundation on genetic simulated annealing algorithm in cloud computing and its accomplishment. HUANG Qi-yi, HUANG Ting-lei proposed [22] a job scheduling strategy and algorithm based on QoS, which could meet user requirements on time and cost. But in the scheduling, the communication between the tasks and the cost of the tasks waiting in the queue are not considered.

## III. OVERVIEW ON GENETIC ALGORITHM

Genetic algorithms are probabilistic Meta heuristic approach, which may be used to explain optimization problems. Genetic algorithms are able to "progress" solutions to real world problems, if they have been properly encoded. It is implicit that a potential solution to a problem may be signified as a set of parameters. These parameters (also considered as genes) are coupled together to outline a string of values (known as a chromosome). The particular values the genes denote are called its alleles. The position of the gene in the chromosome is its locus. Encoding issues handle with signifying a solution in a chromosome. A fitness function must be formulated for each problem to be solved. For a particular chromosome given, the fitness function returns a particular numerical fitness or figure of merit, which will choose the capability of the individual, which that chromosome denotes. It starts with the initial solution called Population and it is filled with chromosome. GA uses Crossover and Mutation operation to produce a new population. By crossover

ISSN: 2321-8169
3867 - 3872

operation, GA produces the neighborhood to explore new feasible solution.

It first forms a primary population consisting of randomly produced solutions. After using genetic operators, namely selection, crossover and mutation, one after the other, new offspring are produced. Conventional view is that crossover is the more momentous of the two techniques for speedily exploring a search space. Mutation offers a small amount of random search, and assists to make certain that no point in the search space has a zero probability of being examined. Then the assessment of the fitness of each individual in the population is done. If the GA has been properly employed, the population will cultivate over successive generations so that the fitness of the finest and the average individual in each generation boosts towards the global optimum. The fittest individuals are taken to be carried over next generation. The above steps are repeated until the termination condition is satisfied. A GA is terminated after a certain number of iterations or if a certain level of fitness value has been reached. After numerous generations, the algorithms converge to the finest chromosome, which expectantly signifies the optimum or suboptimal solution to the problem. The construction of a genetic algorithm for the scheduling problem can be categorized into four parts: The choice of representation of individual in the population; the determination of the fitness function; the design of genetic operators; the determination of probabilities controlling the genetic operators.

**Pseudocode for GA**

***Step 1*** Initial population Generation.

**Encoding**

For applying GAs directly or joined with other meta-heuristics, problem (chromosome) representation is extremely significant and it directly influences the performance of the proposed algorithm. The first choice a designer has to formulate is how to signify a solution in a chromosome.

In GA method, every solution is encoded as a chromosome. Each chromosome has N genes, as chromosome length.

A population is consisting of *chromosomes (or individuals)* and each indicates a possible solution, which is a mapping sequence between virtual machines and physical machines. The initial population can be generated by other heuristic algorithms. For this implementation, each chromosome has number of genes and its corresponding fitness value. One chromosome (or individual) can be represented initially as

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 1 | 4 | 5 | 1 | 2 | 4 | 3 | 1 | 2 |

*Fig. 1*

Here it is considered that if there are 5 physical machines (PM) and ten virtual machines (VM) and virtual machines are to be allocated to physical machines. The initial solution is shown in the figure 1 representing VM1 allocated to PM3, VM2 allocated to PM1, VM3 allocated to PM4, VM4 allocated to PM5 and so on.

***Step 2*** Population Evaluation

Each chromosome is coupled with a fitness value. The aim of GA search is to locate the chromosome with optimal fitness value. For this implementation fitness of individual candidate is calculated by measure of utilized values of physical and virtual machines (generalized, depending on the problem).

Ex. Suppose there are two PMs P1 and P2 having capacity as 50 and 100 respectively.

There are three VMs V1, V2 and V3 having capacity 10, 20 and 30. Supposed all VMs are allocated to P2 then only one PM i.e. P2 is utilized and utilized value will be as

Utilized value = ((10/100)+(20/100)+(30/100))/3

Here Unique count is 1 as one PM used.

$$\text{Fitness} = \text{utilizedvalue} / \text{Uniquecount} \quad (1)$$

***Step 3*** Produce offspring by Crossover.

Crossover operation chooses a random pair of chromosomes and selects a random point in the first chromosome. Roulette Wheel selection operator is considered here for selecting a pair of chromosomes. A crossover operator is used to recombine two strings to get a better string. In crossover operation, recombination procedure generates different individuals in the consecutive generations by combining material from two individuals of the previous generation. In selection of reproduction, better strings in a population are probabilistically allocated a larger number of copies. It is essential to note that no new strings are formed in the reproduction phase. In the crossover operator, new strings are formed by exchanging information among strings of the mating pool. The two strings taking part in the crossover operation are identified as parent strings and the resulting strings are recognized as children strings.

For this implementation crossover operator is applied at the middle of the string.

Ex. Suppose two chromosomes, taken using roulette wheel selection operator, are shown as

X1: {4,2,3,1,2,5,2,1,4,3}

X2: {3,1,5,2,1,3,1,2,5,3}

After performing the crossover, two children are generated as

X1: {4,2,3,1,2, 3,1,2,5,3**}**

X2: **{**3,1,5,2,1, 5,2,1,4,3}

***Step 4*** Performing Mutation to offspring.

Mutation appends new information in an arbitrary way to the genetic search procedure and ultimately supports to avoid getting trapped at local optima. It is an operator that starts diversity in the population whenever the population tends to become uniform due to repeated utilization of reproduction and crossover operators. Mutation may cause the chromosomes of individuals to be diverse from those of their parent individuals. Mutation in a mode is the process of arbitrarily disturbing genetic information.

***Step 5*** *Formation* of the new population for the next generation.

At the end, the chromosomes from this modified population are assessed again. This completes one iteration of the GA. The GA ends when a predefined number of evolutions are reached.

***Step 6*** If terminate condition is arrived finish, otherwise go to Step 2.

## IV.IMPLEMENTATION RESULTS USING GA:

The main goal of scheduling implementation is to schedule virtual machines to the adaptable physical machines (jobs) in accordance with adaptable time, which actually involve finding out an suitable sequence in which all resources can be appropriately utilized.

Scheduling is performed considering various parameters

Implementation considers resource allocation using GA. Three physical machines (i.e. of job types named as a & b, for example) and five virtual machines, each of having different capacity, are taken for scheduling. The main objective is to schedule five virtual machines to the three adaptable physical machines in accordance with adaptable time, which in fact involves finding out a proper sequence in which all virtual machines can be appropriately utilized.

Figure 2 shows number of jobs (two i.e. a and b in this case), number of physical machines (i.e. three) and number of virtual machines (five) (Implementation is done using NetBeans IDE and Java)

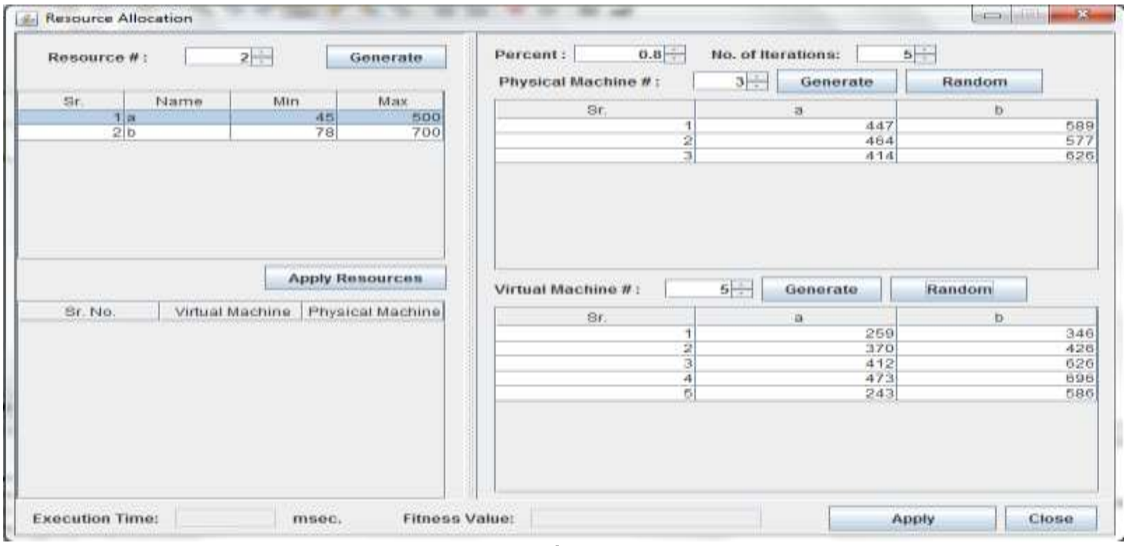


*Fig. 2: Implementation 1*

GA is applied for scheduling virtual machines to physical machines and the implementation result is shown in figure 3. VM1 is allocated to PM2, VM2 to PM1, VM3 to PM1, VM4 to PM1 and VM5 to PM1.

Execution time for this implementation using GA is calculated as 62 ms and fitness value is 0.3551.

Figure 4 shows comment if virtual machines are not allocated to physical machines.

Figure 5 and 6 shows another VM allocation to PM with different fitness value.

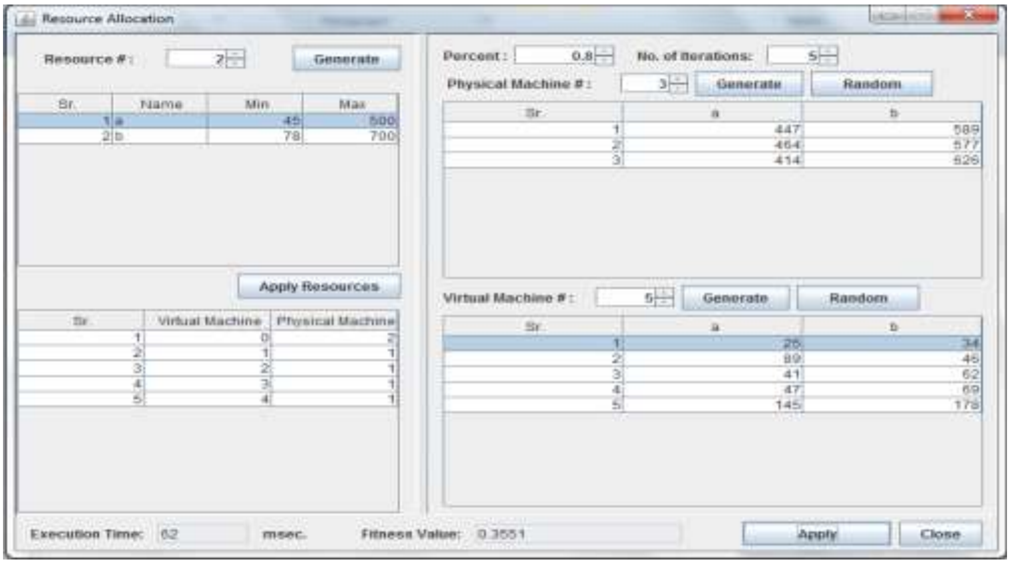


*Fig. 3: Virtual Machine allocation*

 ISSN: 2321-8169
3867 - 3872

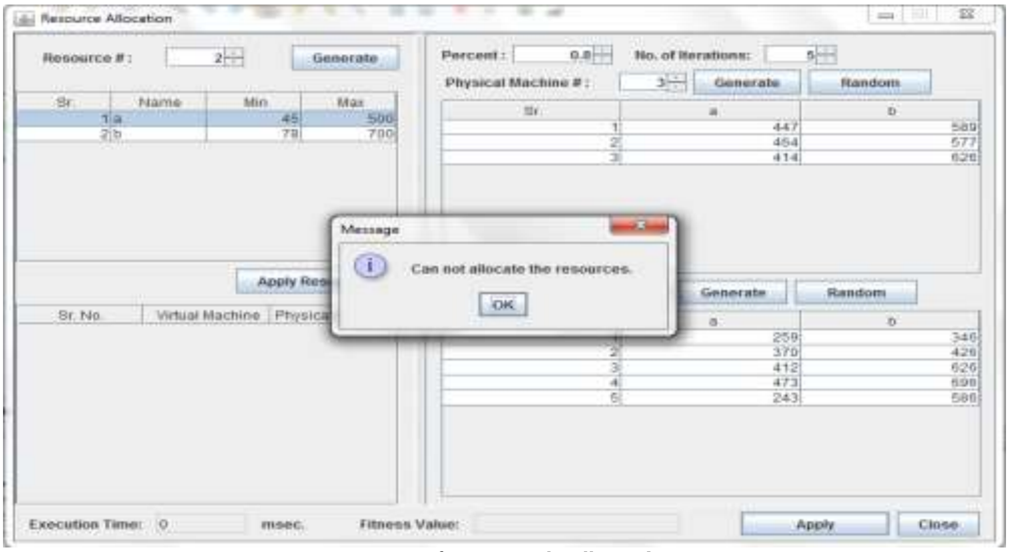


*Fig. 4 : If VM cannot be allocated*

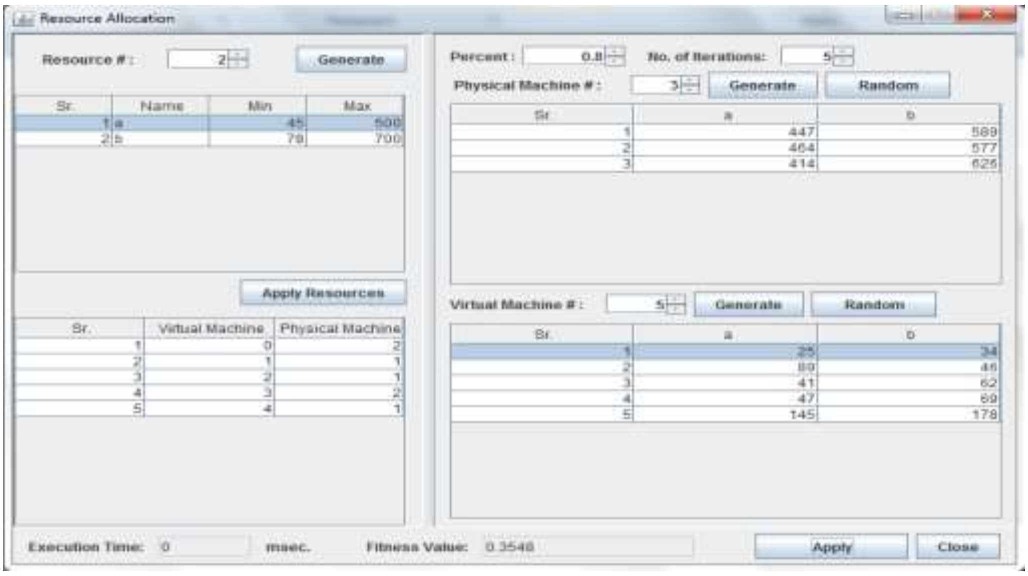


*Fig. 5 : VM Allocation*

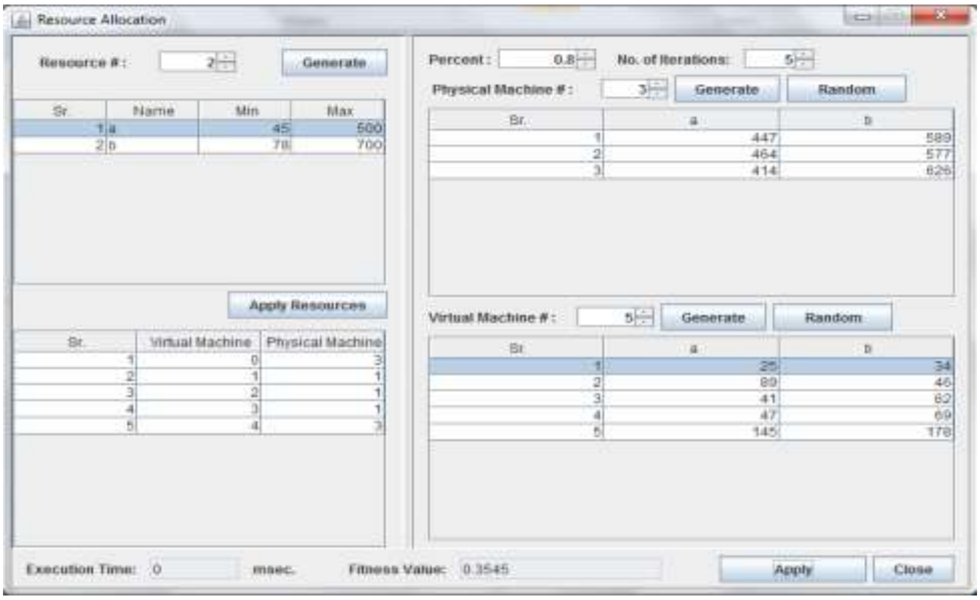


*Fig. 6: VM Allocation*

 ISSN: 2321-8169
3867 - 3872

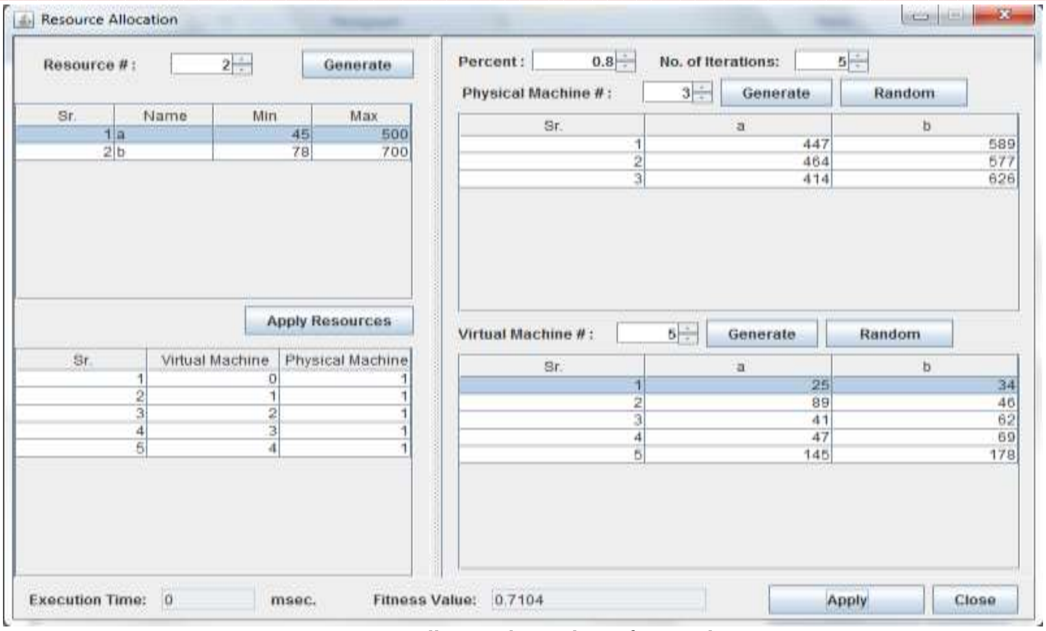


*Fig. 7 : VM Allocation having better fitness value.*

According to the result (figure 7), the execution time required is 0 ms and much better fitness value achieved. Hence all VMs are allocated to PM1 only and others PMs are not used producing better allocation using Genetic Algorithm.

## V. CONCLUSION

In this research work, Job Scheduling is done efficiently using Genetic Algorithm. Different implementations on Genetic Algorithm are also shown and execution times are calculated. Results show that better fitness value produces better schedules as shown in figure 7. Future implementation will consider the hybridization of Genetic Algorithm with any other heuristic algorithm in order to improve execution time and better fitness value.

## REFERENCES

[1] Hamscher, V., et al., Evaluation of Job-Scheduling Strategies for Grid Computing. LECTURE NOTES IN COMPUTER SCIENCE, 2000: pp. 191-202.

[2] Zhiguang, S. and L. Chuang. Modeling and Performance Evaluation of Hierarchical Job Scheduling on the Grids. in Grid and Cooperative Computing, 2007. GCC 2007. Sixth International Conference on. 2007.

[3] Bayati, M., et al. Iterative Scheduling Algorithms. in INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE. 2007.

[4] Patel, Y. and J. Darlington. A Novel Stochastic Algorithm for Scheduling QoS-Constrained Workflows in a Web Service-Oriented Grid. in Web Intelligence and Intelligent Agent Technology Workshops, 2006. WI-IAT 2006 Workshops. 2006 IEEE/WIC/ACM International Conference on. 2006.

[5] Carona, E., V. Garonneb, and A. Tsaregorodtsevc, Definition, modelling and simulation of a grid computing scheduling system forhigh throughput computing Future Generation Computer Systems, 2007. Volume 23(Issue 8): p. 968-976.

[6] Weidong, H., Y. Yang, and L. Chuang. Qos Performance Analysis for Grid Services Dynamic Scheduling System. in Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on. 2007.

[7] Afzal, A., A.S. McGough, and J. Darlington, Capacity planning and scheduling in Grid computing environments. Future Generation Computer Systems 2008. : p. 404-414.

[8] Kiran, M., et al. A prediction module to optimize scheduling in a grid computing environment. in Computer and Communication Engineering, ICCCE International Conference on. 2008.

[9] Rasooli, A., M. Mirza-Aghatabar, and S. Khorsandi. Introduction of Novel Rule Based Algorithms for Scheduling in Grid Computing Systems. in Modeling & Simulation, AICMS Second Asia International Conference on. 2008.

[10] Yuan-Shun, D., X. Min, and P. Kim-Leng, Availability Modeling and Cost Optimization for the Grid Job Management System. Systems, Man and Cybernetics, Part A, IEEE Transactions on, 2008. 38(1): p. 170-179.

[11] Masnida Hussin, Young Choon Lee, Albert Y. Zomaya,"Dynamic Job-Clustering with Different Computing Priorities for Computational Job Alloction", 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010: pp 589-590.

[12] H. Izakian, A. Abraham, and B. T. Ladani, "An Auction method for job allocation in computational grids," *Future Generation Computer Systems*, vol. 26, no. 2, pp. 228-235, 2010.

[13] Di-Wei Huang, Jimmy Lin," Scaling Populations of a Genetic Algorithm for Job Shop Scheduling Problems using MapReduce", 2nd IEEE International Conference on Cloud Computing Technology and Science, 2010.

[14] Attila Csaba Marosi and P´eter Kacsuk," Workers in the Clouds", 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing, 2011: pp 519-524

[15] Prasad and Faruquie and Subramaniam and Mohania, "Job Allocation and SLA Determination for Large Data Processing Services Over Cloud." IEEE ICSC, 2010.

[16] Zelenka, J., Kasanický, T.: Comparison of artificial immune systems with the particle swarm optimization in job-shop scheduling problem. In SAMI 2011: 9th IEEE

ISSN: 2321-8169
3867 - 3872

International Symposium on Applied Machine Intelligence and Informatics. -Budapest : IEEE, 2011, p. 129-135. ISBN 978-1-4244-7428-8.

[17] D. P. Lin and C. K. M. Lee, "A review of the research methodology for the re-entrant scheduling problem," International Journal of Production Research, 2010.

[18] Mrs.S.Selvarani, Dr.G.Sudha Sadhasivam," Improved Job-Grouping Based Pso Algorithm For Task Scheduling In Grid Computing", International Journal of Engineering Science and Technology Vol. 2(9), 2010, 4687-4695

[19] Seonho Kim and Jon B. Weissman, "A GA-based Approach for Scheduling Decomposable Data Grid Applications"

[20] D. Bogdanovich, A. Ziyangirov, "Genetic algorithm structure optimization for the schedule task solution", IEEE 2011

[21] GAN Guo-ning, HUANG Ting-Iei, GAO Shuai," Genetic Simulated Annealing Algorithm for Task Scheduling based on Cloud Computing Environment" IEEE 2010, pp 60-63

[22] HUANG Qi-yi, HUANG Ting-lei," An Optimistic Job Scheduling Strategy based on QoS for Cloud Computing" IEEE 2010 pp 673-675

[23] Yujia Ge, Guiyi Wei," GA-Based Task Scheduler for the Cloud Computing Systems" 2010 International Conference on Web Information Systems and Mining, pp 183-186

[24] Chenhong Zhao, Shanshan Zhang, Qingfeng Liu," Independent Tasks Scheduling Based on Genetic Algorithm in Cloud Computing" 2009 IEEE

[25] Dominique A. Heger," Optimized Job Allocation & Task Scheduling Challenges in Cloud Computing Environments"

[26] Bajeh, A. O., Abolarinwa, K. O.," Optimization: A Comparative Study of Genetic and Tabu Search Algorithms", *International Journal of Computer Applications (0975 – 8887) Volume 31– No.5, October 2011, pp 43-48*

[27] V. Krishna Reddy, 2Dr. L.S.S. Reddy," A survey of various task scheduling algorithms in cloud environment", GJCAT, Vol 2 (1), 2012, pp 847-853

[28] Ajith Abraham, Rajkumar Buyya* and Baikunth Nath," Nature's Heuristics for Scheduling Jobs on Computational Grids"

[29] Salim Bitam," Bees Life Algorithm for Job Scheduling in Cloud Computing",ICCIT 2012, pp 186-191

[30] Jianhua Gu, Jinhua Hu, Tianhai Zhao, Guofei Sun," A New Job Scheduling Strategy Based on Genetic Algorithm in Cloud Computing Environment", JOURNAL OF COMPUTERS, VOL. 7, NO. 1, JANUARY 2012,pp42-52

[31] Mohammad Ranjbar & Mojtaba Najafian Razavi, "A hybrid metaheuristic for concurrent layout and scheduling problem in a job shop environment", Int J Adv Manuf Technol (2012) 62:1249–1260

[32] Fardin Ahmadizar & Mehdi Hosseinabadi Farahani, "A novel hybrid genetic algorithm for the open shop scheduling problem", Int J Adv Manuf Technol (2012) 62:775–787

[33] Hong-an Yang, Qi-feng Sun, Can Saygin & Shu-dong Sun," Job shop scheduling based on earliness and tardiness penalties with due dates and deadline: an enhanced genetic algorithm", Int J Adv Manuf Technol(2012) 61:657-666

[34] Thamilselvan, R. and P. Balasubramanie, "Integration of Genetic Algorithm with Tabu Search for Job Shop Scheduling with Unordered Subsequence Exchange Crossover", Journal of Computer Science 8 (5): 681-693, 2012

[35] Sachin V. Solanki, Dr. A. R. Mahajan, "An Optimized Strategy Based on Genetic Algorithm for Predicting Best Candidate", Elsevier Proceedings of Fourth International Conference on Recent Trends in Information, Telecommunication and Computing – ITC 2013 at Chandigarh, India on Aug 01-02, 2013.